Face Detection

vDetect

Team
J Guna Prasaad
Vipul Venkataraman
Problem Statement

• Given an Image, our program ‘vDetect’ detects and marks the faces in the image.

• Input : Image with faces.

• Output: Image with faces marked.
The solution for the above problem is achieved through a series of processing steps as follows:

- Probability Mapping of Pixels
- Convolution of the Mapped Image
- Thresholding the Convolved Image
- Continuous Component Analysis
- Independent Region Analysis
Probability Mapping of Pixels

- In this step, we are converting the color image to a gray-scale image.

- We are mapping every pixel in the image to a value $p \ (0 < p < 1)$ that denotes the probability that the given pixel value is a face pixel.

- The probability value is derived from a previously created database.

CREATING THE DATABASE:

- We used some training images of 600x800 resolution collected from the internet.

- We decided to operate in the HSV (Hue, Saturation and Value) space as the RGB space is prone to influence by external light.
• We created a database of the number of times each Hue value (least count taken to be 0.1 degrees) has occurred as a face pixel and as a non-face pixel.

• Now, we associate we each Hue value to a number \( p (0 < p < 1) \) which denotes the probability of it being a face pixel value. value ‘p’ is calculated as follows:

\[
\frac{\text{#face-pixel}}{\text{#face-pixel} + \text{#non-face-pixel}^{\text{weight}}}
\]

• Weight that is used in the above formula is to account for the number of times non-face pixel value has occurred as a face pixel.

• Finally, for the input image it is converted to HSV space and mapped to the corresponding probability values from the database.
Convolution of the Mapped Image

- Now, as there seems to be a lot of noise around the face region spotted by the probability mapping, we introduce convolution.

- For every pixel position in the image, consider an enclosing rectangle (of dimensions dependent on the image). Probability of this pixel position is re-mapped to the average of all the probability values in this enclosing rectangle.

- Through convolution, the cases where the region size is less than the rectangle size are eliminated especially in case of single pixel regions.

- After a series of testing and processing we found the optimum ratio of rectangle to image dimensions to be 0.05.
Thresholding the Convolved Image

• Now, after the convolving process, now we set a threshold value for the image (Thresholding value has been experimentally tested).

• Pixel positions with probability higher than the threshold value are left unaltered and others are scaled back to zero.

• This process leads to a better picture of possible faces.
Continuous Component Analysis

- Once thresholding is done, we need to identify those regions with continuous distribution of face pixels.

- For this we enumerate the continuous patches of pixels in each row first. Then, from these list of patches we try to identify continuity with list of patches in row above it.

- We create a rectangle structure which can essentially be visualized as the enclosing rectangle for every continuous region.

- Thus, a list of rectangles are then created using this component analysis.

- But, still we are not sure if these are face patches. They may be any patch associated with the skin color. For example, a combination of faces located close enough to form a continuous patch or patch associated with hand / legs.
Independent Region Analysis

- Given the fact that we have identified continuous patch rectangles, we need to verify if it is a combination of faces.

- To identify this we map every column to a function of number of face pixels in the column.

- From the processed image we are able to see that when there is a combination of face, the above function encounters a local minima.

- Thereby, rejecting some trivial cases we get individual face patches.

- Moreover the cases of hand or legs are eliminated using dimensional analysis
Limitations and Bugs

- Since, the project partially implements machine learning (explicitly though) the precision of the program improves with more training images.

- Moreover, using HSV/RGB space is still critical for the fact that external light will have an influence on the processing.

- As such there is nothing in our project that qualifies to be a bug, but the percentage of success in detecting faces may not equal that of professional softwares.

- Skeleton matching would have led to better independent region analysis.
• Data set needs to be trained more. In the following case the face pixels of some images are not detected due to insufficient training of the database.